The Thesis Committee for Hanyue Zheng
Certifies that this is the approved version of the following thesis:

KKBox Subscription Prediction:
An application of Machine Learning Methods

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor:

Mingyuan Zhou

Timothy H Keitt

KKBox Subscription Prediction:
An application of Machine Learning Methods

by

Hanyue Zheng

Report

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Statistics

The University of Texas at Austin

May, 2018

Dedication

This report is dedicated to my family for their persistent support throughout past years.

Acknowledgements

I would like to acknowledge and thank my supervisor Dr. Mingyuan Zhou for his persistent support and guidance. Thanks to my committee member Dr. Timothy Keitt for the feedback and direction on my report. Special thanks to the Department of Statistics and Data Sciences for offering me the SDS Graduate Fellowship in Fall 2017.

Thanks to my course instructors for inspiring me new perspectives in my study field: Dr. Mary Parker, Dr. Matthew Hersh, Dr. Joydeep Ghosh, Dr. Kam Hamidieh, Dr. Rama T. Lingham, Dr. Michael J. Mahometa. I am also thankful to the program coordinator at Department of Statistics, Vicki Keller, who assisted me on internal program transfer and admission in the summer of 2016.

Thanks to my friends and colleagues at the University of Texas at Austin for supporting me and always being companied: Anna Mengjie Yu, Shiyao Cai, Jieyi Zhu, John Z Li, Zhuoya You, Yanpeng Xi, Jing Xu, Sophia Yang Hooper, Wenhui Zhang, Qi Chen, Qianhui Miao, Xinqi You, Rongsheng Xu, Linli Ding, Yuanrui Ren, Zihan Yang, Qiqi Wang, Yixing Wang.

Special thanks to my friend Jiaqiu Wang who has always been optimistic, supportive and inspiring since the first day we met. It has been amazing to have a friend to share all the moment of excitement, and anxiety during tough times.

These two years will not be such a memorable experience without any of you, I again would like to express my sincere gratitude to all my colleagues for their support and help in the past two years.

Abstract

KKBox Subscription Prediction: An application of Machine Learning Methods

Hanyue Zheng, M.S. Stat.

The University of Texas at Austin, 2018

Supervisor: Mingyuan Zhou

This report used datasets from a Kaggle competition which aims to develop machine learning models to predict if users of a music app called KKBox will renew their membership after it expires. This report created four machine learning classification models including logistic regression, random forest, Naïve Bayes and gradient boosting. Exploratory data analysis was performed to understand data distribution and the relationships between features. For models cannot handle missing data and multicollinearity, data imputation and principle component analysis were performed. The result shows that the variable importance derived from models are quite different, which suggests us to be more cautious selecting models. It is also shown that the random forest model achieved the highest AUC (0.9727), followed by Xgboost (AUC = 0.0921), logistic regression (AUC = 0.8500), and Naïve Bayes (AUC = 0.7962). However, it is unrealistic to judge model performance without considering the real business case. The result from this report is a guidance for further business decision making.

Table of Contents

List of Tables

List of Figures

Introduction

BACKGROUND

This report was inspired by a Kaggle competition named WSDM – KKBox's Churn Prediction Challenge (https://www.kaggle.com/c/kkbox-churn-prediction-challenge). This competition aims to build up several predictive models to help determine whether a user will churn after their subscription expires. This report will use the dataset provided by the competition.

KKBox is a leading music streaming service in Asia. Offering millions of people access to their music library by advertising and paid subscription, KKBox sees their paid users make up an important proportion of the profits. At present, KKBox is using survival analysis technique to determine subscriber's remaining membership life time. Therefore, it is not only essential to develop a model with accurate prediction of churn of their paid users, but also important to understand why users leave or stay so that the company can make better decisions and promotions to increase user retention. New ideas and algorithms are expected to be implemented to help them solve those issues.

At present, machine learning techniques have been applied to various research areas including natural science, social science, and engineering. Those techniques drew industry attention in recent years, because the power and accuracy of machine learning algorithms help the industry save money. This report is a perfect opportunity to apply machine learning algorithms to real life data and make recommendations on the business strategy. Another advantage of using machine learning methods is that we are able to ensemble multiple models to build up a sophisticated but accurate forecasting model.

Machine learning algorithms can be generally divided into two types based on data structure: supervised learning and unsupervised learning. In terms of the problem

type, machine learning algorithms can be used to solve two main types of problems: regression problems and classification problems. In this report, we will focus on the supervised classification algorithms. A classification algorithm is used to identify a variable with a set of categories. The mostly used methods include logistic regression, support vector machine, K-nearest neighbor, random forest (decision trees), Naïve Bayes, and gradient boosting. Logistic regression and Naïve Bayes are linear classifiers. The advantage of using linear classifier is that they are usually trained fast. Support vector machine usually performs well with small datasets with no missing data. K-nearest neighbor is a widely used multi-classification algorithm, but it might perform slowly for large datasets. Random forest and gradient boosting are ensemble methods and usually more stable and powerful.

This report selected four models for training, two are linear classifiers (logistic regression and Naïve Bayes), two are ensemble methods (random forest and gradient boosting). The final output will be the accuracy, recall and precision rate of each model.

DATASETS

Four datasets ('train', 'transaction', 'user log' and 'member') stored in .csv format were downloaded from Kaggle KKBox competition homepage. The 'train' dataset includes a column called 'msno' which is a unique identifier of each user, and a binary response variable 'is_churn' where 1 indicates the user did not renew after subscription expired, and 0 indicates the user did. The 'transaction' dataset comprises all the information about user's transaction on KKBox such as payment type, payment id, transaction date, and if a user chose to cancel or auto-renew the subscription, etc. The 'user log' data include user behavior data on the app, such as counts of songs that played completely, the number of seconds played by each user, and the percentage of songs a

user listened till they switched (0 - 25%, 25 - 50%, 50 - 75%, 75 - 98.5%, and 98.5 - 100%). The 'member' dataset provides user information including city, age, gender, registration method and registration date. Table 1 lists all the features and a brief explanation. There is lots of missing data and the summary of it will be discussed later.

This report will not use the 'test.csv' provided on Kaggle. The merged train set will be split into 80% as train and 20% as test sets. Since the test set contains response variable, we will be able to validate and compare model performances.

Train.csv	msno	A unique identifier indicating each user
	is_churn	Target variable
Transac-tions.csv	payment_method_id	Payment method
	payment_plan_days	Length of membership plan in days
	plan_list_price	In New Taiwan Dollar (NTD)
	actual_amount_paid	In New Taiwan Dollar (NTD)
	is_auto_renew	Whether the user auto-renew the subscription
	transaction_date	Format %Y%m%d
	membership_expire_date	Format %Y%m%d
	is_cancel	Whether or not the user canceled the membership in this transaction
User_logs .csv	date	Format %Y%m%d
	num_25	# of songs played less than 25% of the song length
	num_50	# of songs played between 25% to 50% of the song length
	num_75	# of songs played between 50% to 75% of the song length
	num_985	# of songs played between 75% to 98.5% of the song length
	num_100	# of songs played over 98.5% of the song length
	num_unq	# of unique songs played
	total_secs	Total seconds played
Members. csv	city	City name
	bd	Age
	gender	User's gender
	registered_via	Registration method
	registration_init_time	Format %Y%m%d

Table 1. Explanation of all the variables used in the report (variable demonstration revised from Kaggle competition site, https://www.kaggle.com/c/kkbox-churn-prediction-challenge/data).

Methods

DATA PRE-PROCESSING

Data preprocessing includes merging and manipulating data from four datasets, performing data visualization, imputing new variables, selecting features using principle component analysis (PCA), and splitting data into train and test datasets. Packages used in this process include 'dplyr', 'tidyr', 'ggplot2', 'corrplot', etc. R packages such as 'data.table' are used to speed up data reading. To save memory and increase efficiency, we only randomly select 100,000 rows from the dataset instead of all the data for exploratory data analysis and modelling.

Data manipulation is essential for this project as data were split in separate excel sheets. Datasets are merged by a unique column, 'msno'. Unrealistic and outlier data will be removed to develop a representative and more general model. New variables can be created by extracting useful information from current data or external resources.

Data visualization aims to create various plots so that we can easily tell the distribution of a feature or the relationships between features. Plots involved include histogram, boxplot, scatter plot, correlation plot, etc. Data visualization is a convenient way to make a first judgement on the importance of variables in prediction. Besides, model assumptions can be easily checked using graphs.

It is essential to check multicollinearity before developing the model. Some machine learning models, random forest and gradient boosting for example, can handle multicollinearity well. However, it will be a problem in prediction using logistic regression and Naïve Bayes. PCA will be used to resolve this issue and reduce data dimension and complexity when features have strong correlations. The dataset will be split into training and test data at the ratio of 8 to 2.

MODELING METHODS

Four machine learning models were developed in this project: logistic regression, random forest, Naïve Bayes, and gradient boosting.

Logistic regression is a binary classifier developed in 1958 (Cox, 1958). It is widely used for predicting binary dependent variable based on one or more features. The logistic function is written as below:

$$P(y) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x + \cdots + \beta_n x)}}$$

Where n is the number of features. The advantage of logistic regression is that it can be fast trained, however, it cannot deal with missing data.

Random forest is an ensemble method using bootstrap aggregating which integrates multiple decision tree models. This algorithm was firstly created by Tin Kam Ho (1995). Random forest can be used for classification or regression.

Each individual tree employs a structure of decision nodes and branches. The tree model starts with a root node, and each internal node is a binary split labeled with an input feature. Decision tree is a greedy algorithm because it finds the best split at each step by finding the best improvement. It computes a value between 0 and 1 as the predicted probability. Given certain threshold, the probability is classified into one of the desired outcome. It is essential to try different threshold to find the best tradeoff between false negative and false positive rate. In order to evaluate misclassification, Gini index (Breiman et al., 1984) for each node will be calculated to evaluate the purity:

$$p_1(1 - p_1) + p_2(1 - p_2)$$

Where p_1 and p_2 are the probability of class 1 and class 2 at the node, respectively. The train sets for each tree are subsamples taken with replacement from the original dataset. Each subsample set is used to build one decision tree. The bagging

method can reduce model variance without increasing bias. Besides, noise can be reduced if trees are not correlated.

For $b = 1, \dots, B$:

1. Sample with replacement, n training examples from X, Y; call these X_b, Y_b.

2. Train a classification or regression tree f_b on X_b, Y_b.

Naïve Bayes is a simple classification method which constructs a conditional probability model using Bayes' Theorem. It assumes that the effect of the value of each feature on a given class is independent from the value of all the others (class conditional independence). Suppose we have n features, the conditional probability of kth feature can be decomposed as:

$$p(C_k|x) = \frac{p(C_k)p(x|C_k)}{p(x)}$$

Where C_k is the outcome or class of kth feature, $p(C_k)$ is the prior probability of class k, $p(x|C_k)$ is the likelihood that the probability of predictor given class, and $p(x)$ is the prior probability of the predictor. The main aim of the algorithm is to calculate the conditional probability of a feature vector X with class k, which becomes:

$$p(C_i|x_1, x_2, \dots, x_n) = \frac{p(C_i)p(x_1, x_2, \dots, x_n|C_i)}{p(x_1, x_2, \dots, x_n)}, 1 \leq i \leq k$$

Gradient boosting is a machine learning method raised by Leo Breiman (Breiman, 1997) which ensembles multiple weak prediction models such as decision trees to produce a strong classifier. It is the combination of gradient descent and boosting and was evolved from AdaBoost to stochastic gradient boosting (Friedman et al., 2000). Gradient descent takes steps to the proportionally negative gradient of the function from

the current point to find the local minimum. Boosting is a machine learning ensemble algorithm used to reduce bias and variance.

Boosters are learned sequentially with early learners fitting simple models to the data and then analyzing the data for errors. Those errors identify problems or particular instances of the data that are hard to fit examples. Then later models focus primarily on those hard examples trying to get the prediction right. All the models will be given weights in the end as an overall predictor. Therefore, boosting is a technique converting a sequence of weak learners into a complex predictor.

In this report, we choose to use Xgboost which is one of the mostly used gradient boosting methods. It was initiated by Tianqi Chen and soon become famous as a machine learning method after he won the Higgs Machine Learning Challenge. Xgboost is an implementation of gradient boosting that can significantly improve execution speed and model performance.

Results

Exploratory Data Analysis

In this session, multiple plots will be used to display features distribution and the relationships between features. Histogram and density plot are used to show the distribution of continuous numeric data, and barplot is used to display frequency counts of each levels. Given the large number of features, only selected features will be shown to explain the critical relationships.

The response variable 'is_churn' is a binary variable where 1 indicating the user did not renew after subscription expired, and 0 indicating the user did. It is clear that this dataset is imbalanced. Only 6.39% users chose to leave after their subscription expired.

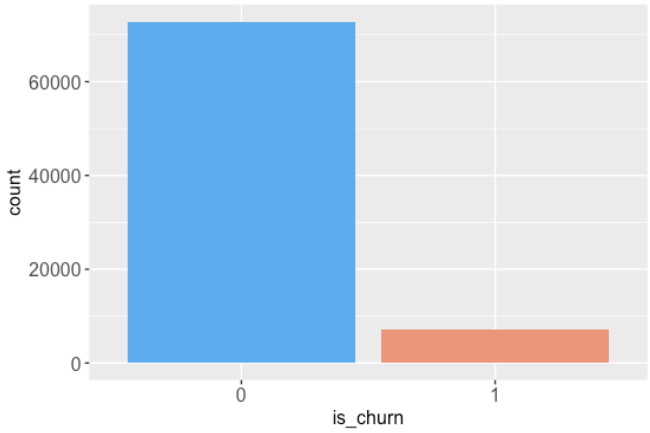

Figure 1. Barplot showing frequency counts of the response variable 'is_churn'.

Demographically, the users' age data has outliers that -974 as the lower bound and 1820 as the upper bound. To make the graph more readable, age smaller than 0 and larger than 100 are removed for this visualization. Users' age is right skewed with mean 30 and median 28. Users' location data is dummy coded as numbers. 65.08% of users

come from city 1. The percentage of users who churn or not churn follow the same pattern among all cities which indicates that city might not be a significant effect to predict membership renewal.

It is also noticeable from the visualization that the majority of users (~600k) do not provide gender information. There is no gender difference for users who choose not to continue their service (Figure 3). However, we see majority of staying users have no gender information. Male users are more inclined to renew.

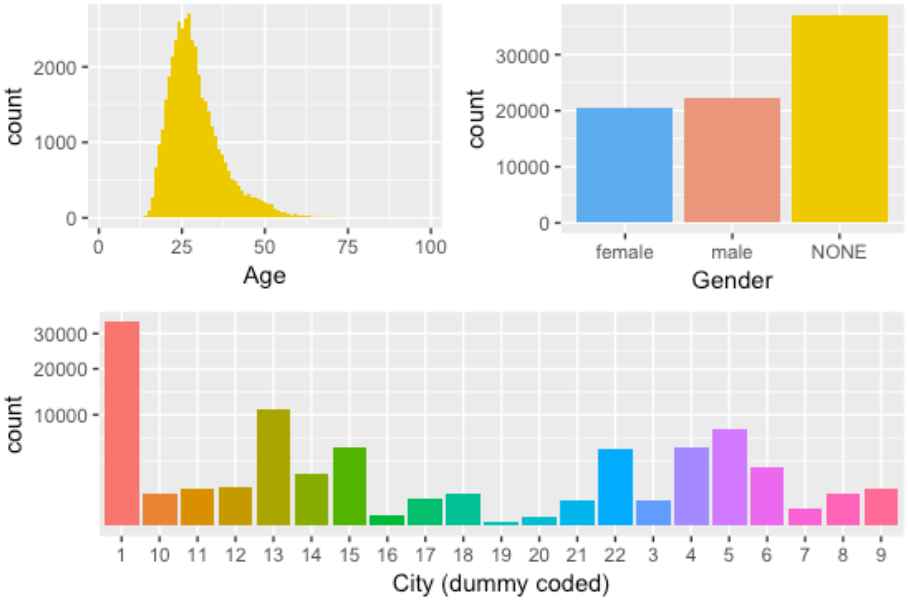

Figure 2. User demographics summarized from 'member' dataset.

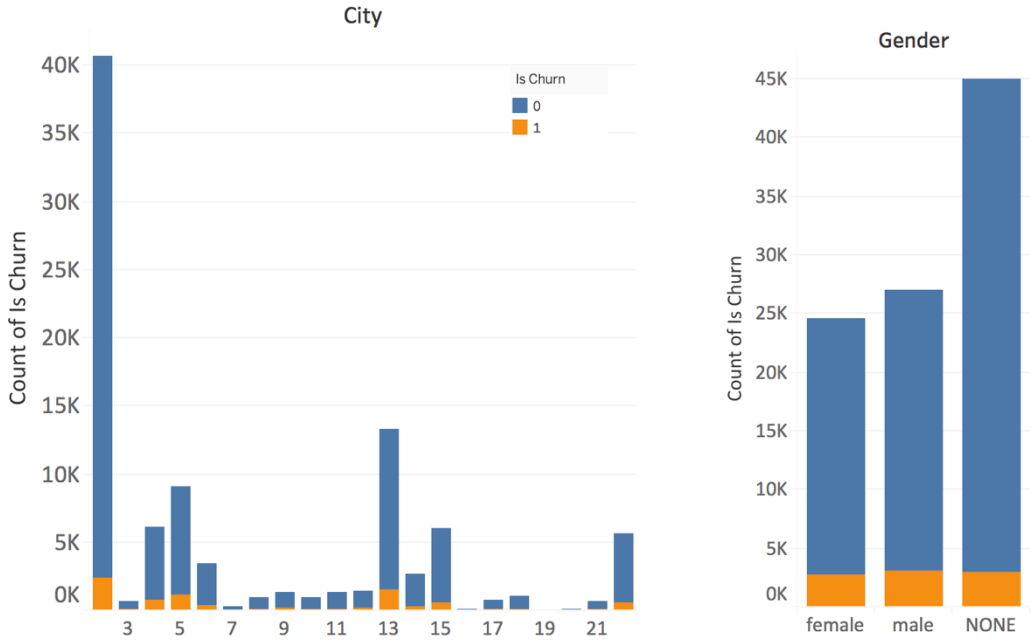

Figure 3. Relationships between city, gender and the target variable.

Originally, there are nineteen different registration methods dummy coded. Our dataset only contains five of them which also indicated that those methods must be dominant among the population. Among the five methods, method 3, 4, 7, 9 have much higher counts. 48.64% users prefer payment method 41 for making transactions. The majority users have their subscription automatically renewed and do not have subscription canceled. Payment plan days can last up to 450 days, but are mostly around a month (30 or 31 days). The payment plan price and actual paid amount are very similar, represented by a straight diagonal line in the figure. There are outliers such as plan price is greater than 0 but actual payment amount is 0, indicating that those users did not renew the subscription after it expired.

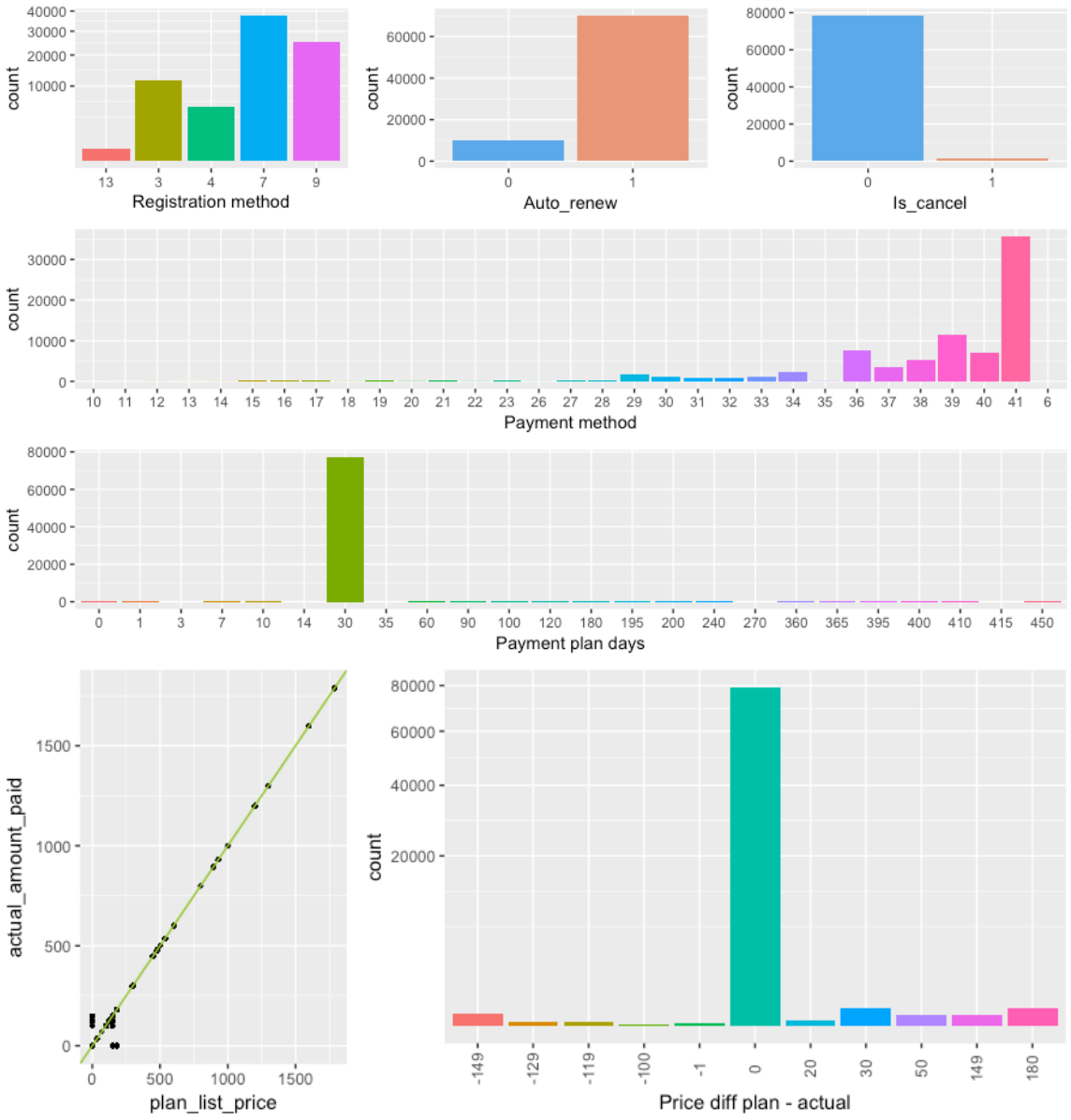

Figure 4. Plots showing distributions of features, and relationships between features in dataset ‘transactions’.

In the ‘user log’ dataset, the distribution of percentage of song completion are very similar except the 100% completed songs. This implies that most of the time people skip the song at the beginning. It makes sense that unless people like the song and want to

listen till the end, they could skip it any time before it ends. Besides, completed songs have significant broader count range than the others. The number of unique songs played per day shows an exponential shape, and the distribution of total second played is left skewed.

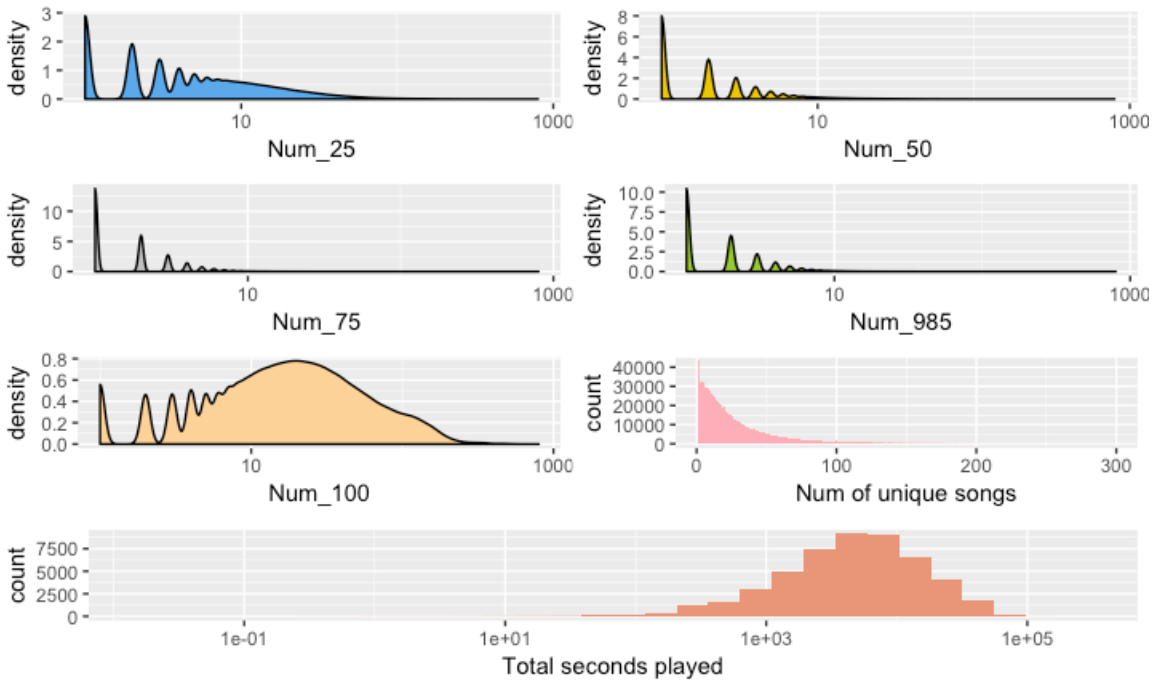

Figure 5. Plots showing distributions of features, and relationships between features in dataset 'user log'.

DATA PREPROCESSING

All the datasets were joined by column 'msno' to construct the final dataset which contains 16,887,877 rows and 23 columns among which seven of them are categorical variables, and the rest of them are numeric.

Firstly, we identified the missing values under each feature. Ensemble models such as random forest and gradient boosting can handle missing values. Logistic regression, however, is more sensitive to missing data. Data imputation is required to improve model performance. By summarizing the missing data counts of each feature (Table 2), the features coming from the same dataset have the same number of missing values. It makes sense because datasets are joined using the same column, 'msno'. If one user's id is mismatched or missing, then all corresponding columns will be null value. Considering the missing data only count for about 2% of the total data, we can simply neglect this portion and only use the complete non-missing data for model development and prediction.

Msno	is_ churn	city	bd	gender	registered_ via	registration _init_time	payment_ method_id
0	0	117166	117166	117166	117166	117166	340641
payment_ plan_days	plan_lis t_price	actual_am ount_paid	is_auto _renew	transacti on_date	membership_ expire_date	is_cancel	date
340641	340641	340641	340641	340641	340641	340641	249615
num_25	num_50	num_75	num_9 85	num_10 0	num_unq	total_secs	
249615	249615	249615	249615	249615	249615	249615	

Table 2. Summary of missing data of all the features.

In R, we used function set.seed() and plugged in value 10,000 to select 100,000 sample randomly from the original data. 96,556 rows were left after removing the missing data. 80% are used as training set, the rest are used as test set.

From the exploratory data analysis, we have already knew that the age of users (column 'bd') can range from -974 to 1820. In reality, people's age is mostly smaller than 100, therefore we remove outlier values out of range 0 to 100. This dataset also contains

various time series data such as: initial registration time, transaction date and membership expiration date, date. Those dates are not continuous numeric data, and cannot be treated as categorical variable which may result in too many dummy variables. We instead extracted the year, month, and weekday information from the dates to preserve more time information and drop the dates.

The correlation plot was created to show the correlation between numeric features (Figure 5). It is shown that some features are strongly correlated with each other. Week of the transaction and year of the transaction are highly correlated (coefficient -0.84). It is reasonable to find that features from the same dataset are more related, such as num_50 and num_75, indicating that users who listen to 50% of a song might end up with keeping listening till the 75%.

We therefore created scatters plot to further identify their relationships (Figure 6). As we can see, date_week and date_mday are very highly correlated. It might be a result of the column 'date' has all seven days in a week, so that they are completely correlated. It also makes sense that the transaction day and membership expiration day are correlated. For users who have no auto payment set up, they might only remember to pay the fee at the last minute.

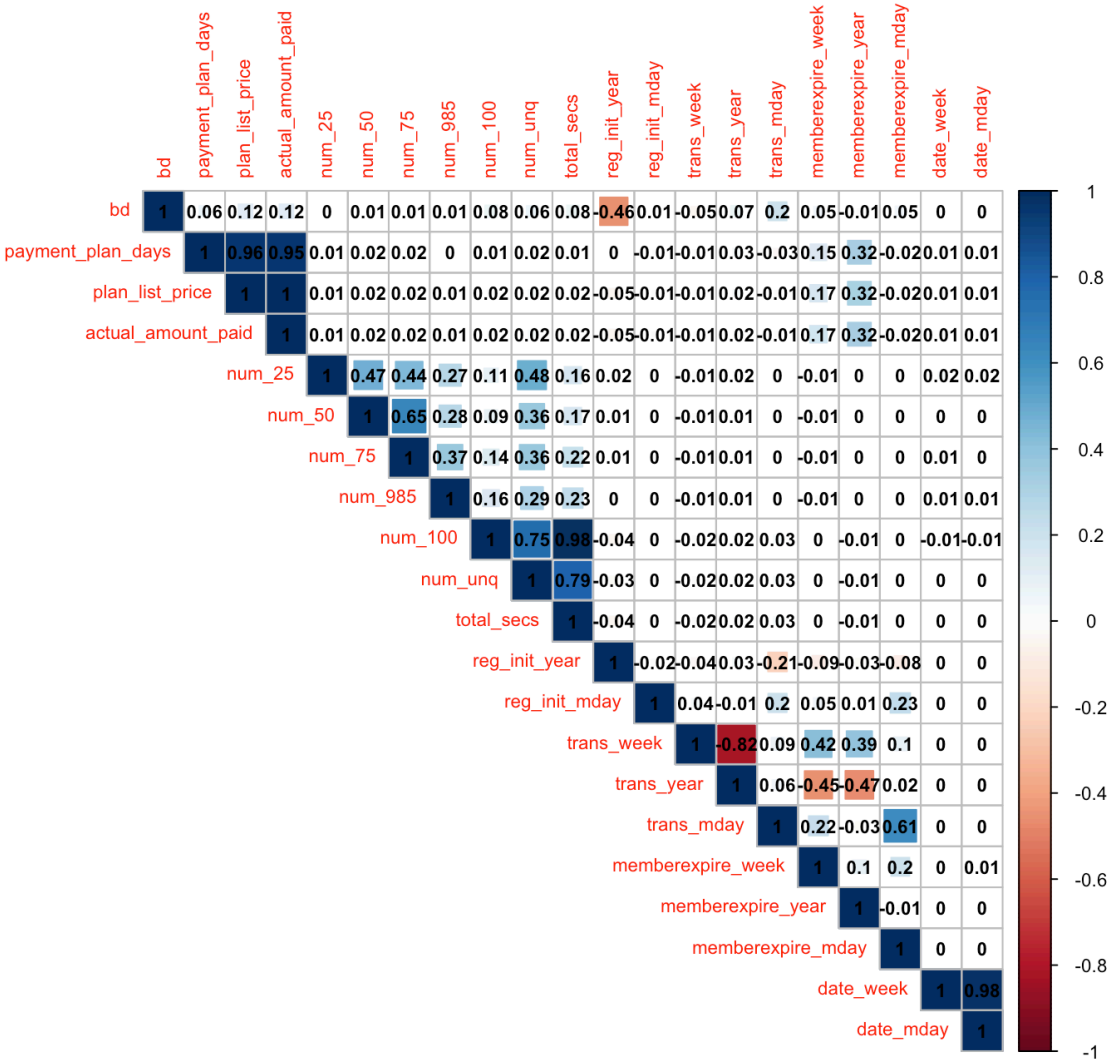

Figure 6. Correlation coefficients between numeric features.

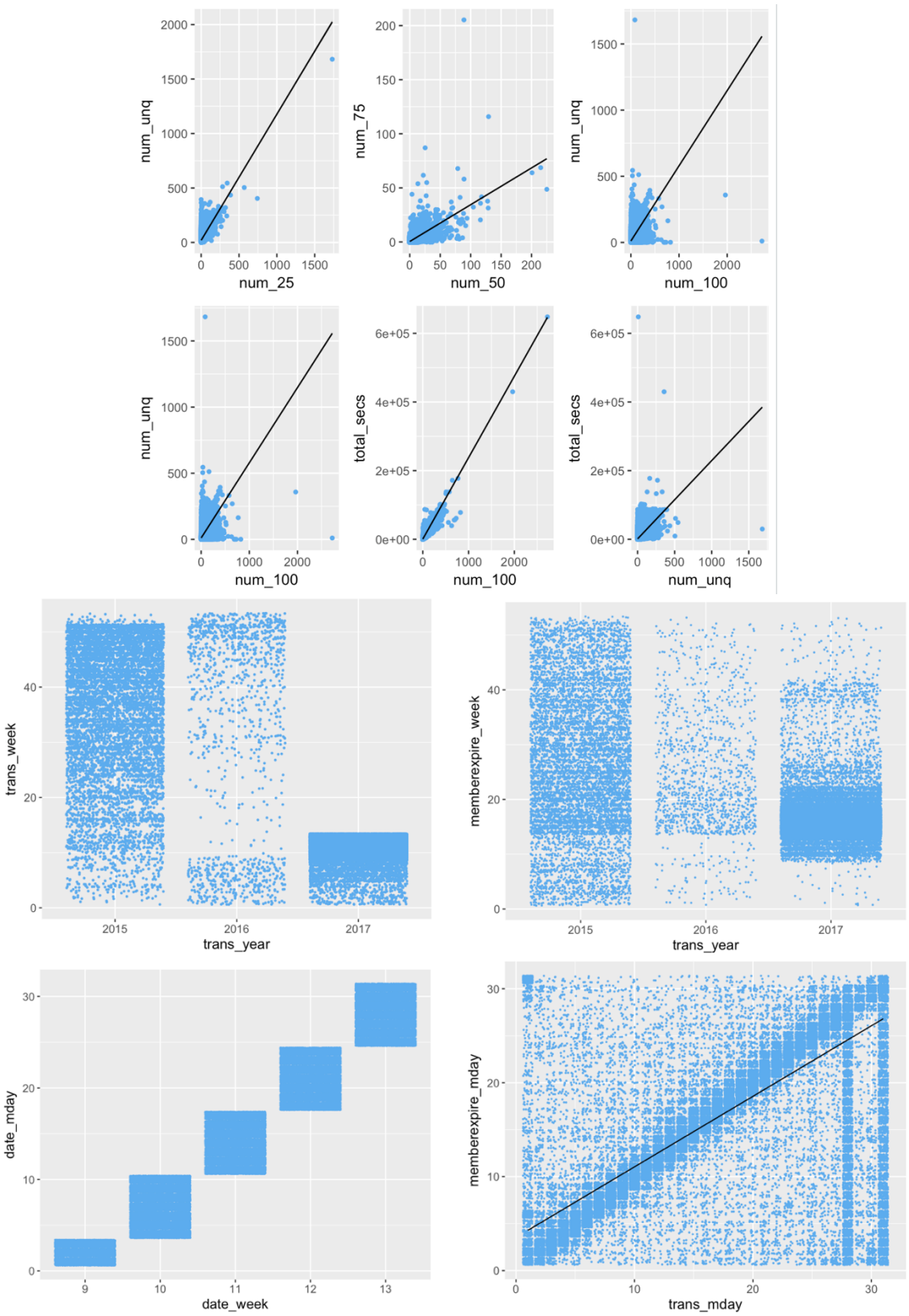

Figure 7. Scatter plots showing relationships between highly correlated features.

PCA was performed for dimension reduction and resolving multicollinearity for highly correlated data. Eight features associated with correlation greater than 0.8 were selected for PCA: num_50, num_100, total_secs, num_unq, trans_week, trans_year, trans_mday, memberexpire_mday. All of these features are numeric so no dummy coding is needed. Figure 8 shows the percentage of variance of eight features. Table 3 shows that the top four features take account 91.71% variation. In this project, we will pick the top four principle components for logistic regression and Naïve Bayes.

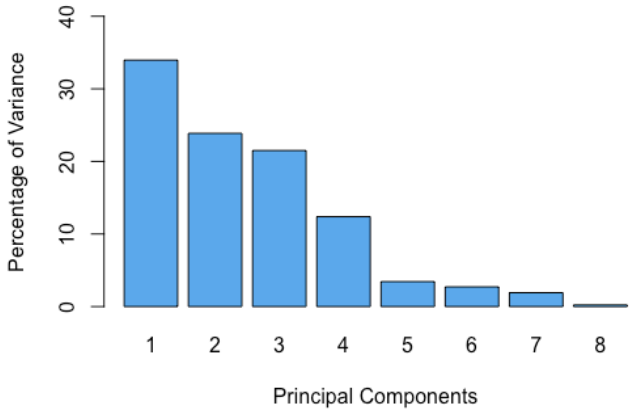

Figure 8. Percentage of variance of all principal components, including dummy coded categorical variables and numeric variables.

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
Standard deviation	1.6482	1.3818	1.3116	0.9952	0.5246	0.4662	0.3907	0.1337
Proportion of Variance	0.3396	0.2387	0.2150	0.1238	0.0344	0.0272	0.0191	0.0022
Cumulative Proportion	0.3396	0.5783	0.7933	0.9171	0.9150	0.9787	0.9978	1.000

Table 3. Summary of standard deviation and variance proportion of the principle component analysis.

VARIABLE IMPORTANCE INTERPRETATION

We are able to get the rank of variable importance from our models. Logistic regression does not give comparable variable importance directly because the coefficients are not standardized. However, we can compare the Z-score as well as the P-value. The larger the Z-score and the smaller the P-value, the more importance the variable is. We found out that whether the user canceled the membership is the most important feature. Registration year, week of membership expiration, and whether the user set up auto renew are also significant (absolute value of Z-score is greater than 10). This result shows that if a user chooses to cancel membership during a transaction, or pays more during a transaction, it is more possible that this person will not renew the membership after it expires.

	Estimate	Standard Error	Z-score	P-value
is_cancel0	-2.936e+00	5.402e-02	-54.350	< 2e-16
reg_init_year	-3.056e-01	7.928e-03	-38.544	< 2e-16
memberexpire_week	9.274e-02	2.611e-03	35.517	< 2e-16
is_auto_renew0	1.485e+00	7.732e-02	19.200	< 2e-16
actual_amount_paid	-1.660e-02	1.018e-03	-16.301	< 2e-16
payment_plan_days	7.984e-02	4.991e-03	15.999	< 2e-16

Table 4. Estimated value, standard error, Z-score and P-value of selected variables created by logistic regression model.

Random forest model also measured variable importance by mean decrease accuracy. Figure 9 shows that the timing (both week and year) of transaction is the most significant feature. Payment plans and the payment amount play important roles in

prediction. A user who decides to pay for a longer period membership for larger amount of money is more possible to renew the membership. Similar to logistic regression, the week that membership expires are important as well. This model also indicates that the initial registration time and the percentage listening to a song also have effects on churn rate.

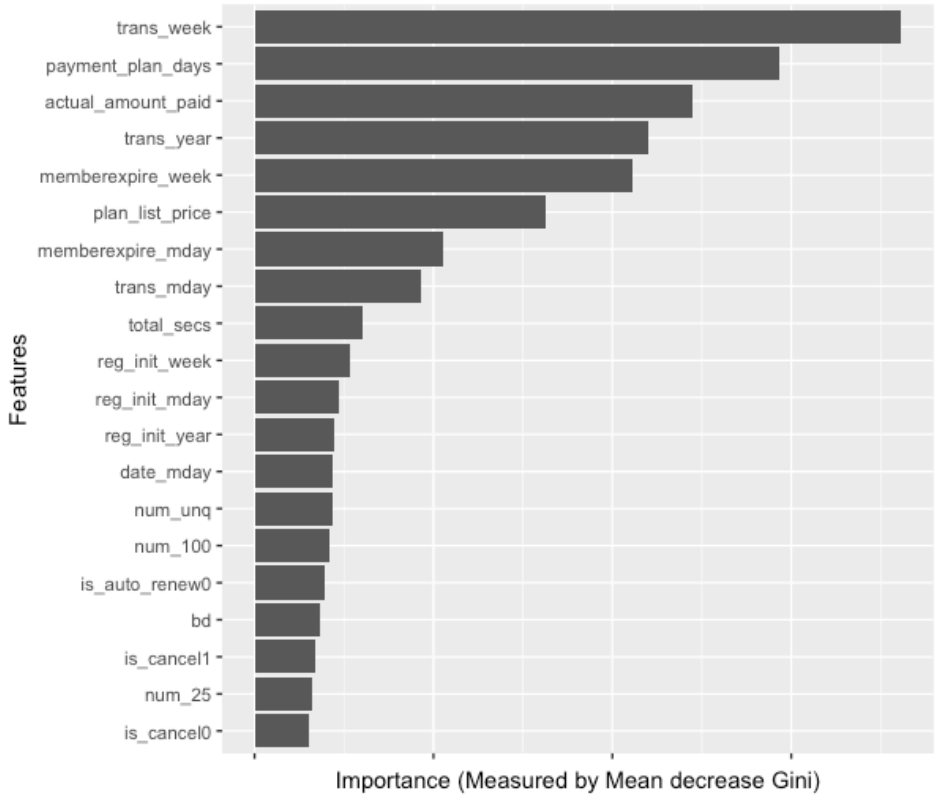

Figure 9. Top 20 variable importance for random forest model.

Gradient boosting can also measure variable importance. The package ‘xgboost’ contains a function named ‘xgb.importance’ which generates a column ‘gain’. ‘Gain’ represents the improvement in accuracy brought by a feature, in other words, the contribution to the model. Ideally, we expect a similar ranking as random forest does

although the variable importance is measured differently. However, gradient boosting suggests that demographics (age and gender) are the most important feature, followed by registration method and payment method. This result is completely different from logistic regression and random forest.

To sum up, although we expect different models to have similar rank for variable importance, they could actually differ significantly. We should be cautious about choosing a model as well as using the domain knowledge to justify out choice.

Figure 10. Top 20 variable importance for gradient boosting model.

PREDICTIVE MODELING PERFORMANCES

Model performance is compared and visualized by the ROC curve. ROC is short for receiver operating characteristic, it is a graphical plot that displays the diagnostic

ability of a binary classifier. The two attributes are sensitivity and specificity. Sensitivity refers to true positive rate, and specificity represents 1 - false positive rate. The ROC plot is developed using false positive rate (1 - specificity) against true positive rate (sensitivity). Ideally, we would like a steep trajectory reaching 1 on the left side of the graph. Among the four models, random forest achieves the highest AUC (0.9727), followed by Xgboost (AUC = 0.0921), logistic regression (AUC = 0.8500), and Naïve Bayes (AUC = 0.7962).

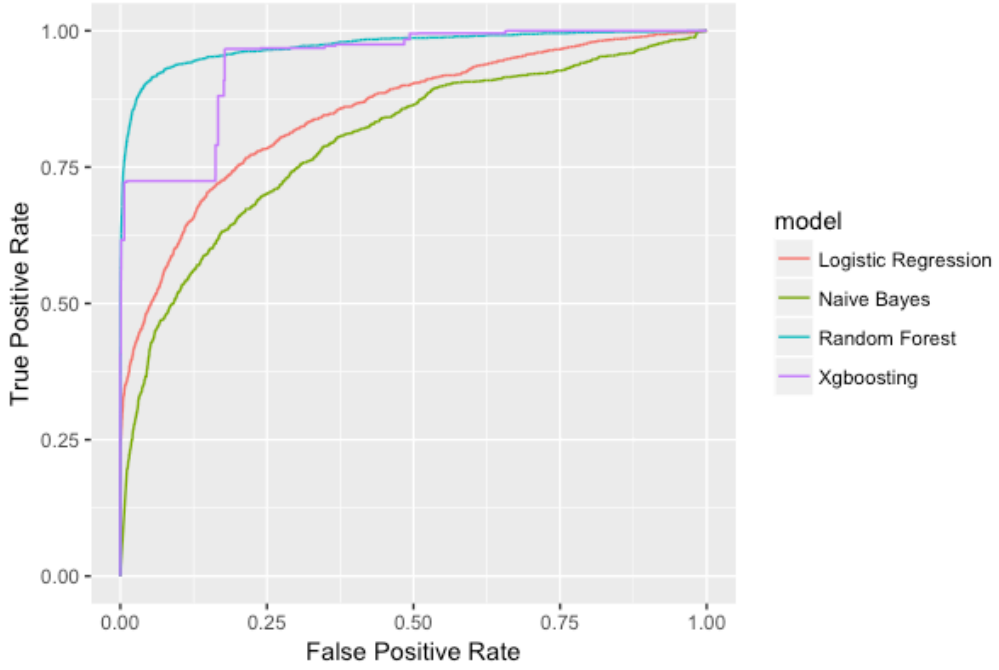

Figure 11. ROC curves of four models.

In addition, confusion matrices from models with high AUC were reported (Table 5). We can see that both models provide similar precision and recall rate. The random forest model is more in favor because it predicts the most negative case correctly and also maintains higher true positive prediction rate. From a business perspective, this study

does emphasize on higher recall rate because we want to know who exactly will not renew the membership. But if the precision is too low, too much effort might be spent on the negative cases. Even though we can choose a cutoff based on the accuracy rate, it is essential to think about the real business to decide the best model and parameters.

Random Forest:		Observed	
		0	1
Predicted	0	14951	360
	1	99	1146

Xgboost:		Observed	
		0	1
Predicted	0	14948	423
	1	102	1083

Table 5. Confusion matrices reported from random forest and Xgboost models at the cutoff of best accuracy.

Conclusion

This report used data from a Kaggle competition to perform a real-life application of machine learning models. Data was summarized by histograms, bar plots and scatter plots to show the relationship between features. Summary statistics was also created for variables of interest. It was shown that this dataset is extremely imbalanced where only 6.39% data hold negative outcome, and the rest are positive. Besides, from the explanatory data analysis we can know that most users prefer only one payment method, and come from the same city. For an imbalanced dataset, it is important to tune the parameter to find the best prediction balance between positive and negative outcomes (between precision and recall rates). In our models, the default parameters actually worked fine. Parameters were only tuned for the Xgboost model. The AUC showed that random forest model performs the best, followed by Xgboost, logistic regression and Naïve Bayes. Confusion matrix also suggested that random forest can give good prediction for both positive and negative cases.

In terms of variable importance, logistic regression and random forest provide similar results, where variables such as payment amount, payment plan days, is_cancel, initial registration time, is_auto_renew are significant. However, Xgboost gives a completely different ranking. User age, gender and registration method are the most important features. It is noticeable that more than half of the users do not fill out their gender. Since gender seems to be an important feature for the Xgboost model, it will help improve the prediction by imputing missing gender information.

In conclusion, developing different predictive models might help us understand the business and judge for the best model for production. There could be a trade-off between training time and descent prediction. For example, although random forest

performs the best, it is trained for much longer time than the other models. It is our call to find the most appropriate model for real business.

References

Bishop, C. (2011). Pattern Recognition and Machine Learning (Information Science and Statistics).

Boosting (machine learning), https://en.wikipedia.org/wiki/Boosting_(machine_learning). Retrieved on March 25th, 2018.

Breiman, L. (1997). "Arcing The Edge".

Chen, T. (2016). "Story and Lessons behind the evolution of XGBoost".

Gradient descent, https://en.wikipedia.org/wiki/Gradient_descent. Retrieved on March 25th, 2018

Ho, T. (1995). Random Decision Forests. Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14–16 August 1995. pp. 278–282.

Jerome F., Trevor H., and Robert T. (2001). The elements of statistical learning, volume 1. Springer series in statistics Springer, Berlin.

Jollie I. (2002). Principal component analysis. Wiley Online Library.

Random forest, https://en.wikipedia.org/wiki/Random_forest. Retrieved on April 9th, 2018.